

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-204966

(43)Date of publication of application : 31.07.2001

(51)Int.Cl.

A63F 13/10

(21)Application number : 2000-018294

(71)Applicant : NINTENDO CO LTD

(22)Date of filing : 27.01.2000

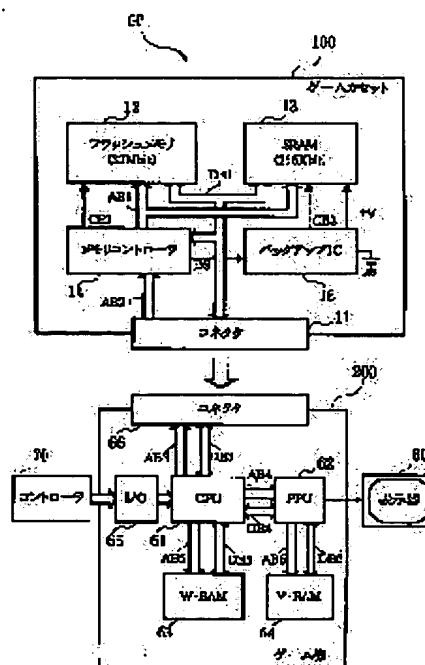
(72)Inventor : IIJIMA YUTAKA  
 MATSUNADA TAKUMA  
 OHARA TAKAO  
 YOSHINO MOTOFUMI  
 YOSHIOKA TERUYUKI  
 ARAI SHIGENOBU  
 HOTTA TAKUJI  
 HIRAI KAZUICHI

## (54) GAME APPARATUS AND INFORMATION STORING MEDIUM FOR GAME

## (57)Abstract:

PROBLEM TO BE SOLVED: To provide a game apparatus wherein the development of a game is changed more in accordance with progressive conditions of a plurality of games, and an information storing medium storing game programs.

SOLUTION: In the apparatus (GP) with which a plurality of mutually related games can be played, a flash memory (12) stores a plurality of the mutually related game programs (PG) and an SRAM (13) writably and readably stores a back-up data (DG) expressing the progressive condition of each of individual games (A, B and C) and a player selects one among a plurality of the game programs (PG) by means of a controller and indicates starting and advancing of the game and a judging unit (61) judges the existence of back-up data DG of a related separate game when it is instructed to start the game and game program changers (S234-S262) change a part (a3-c7) of the game programs PG and S based on the result of judgement in accordance with the back-up data (DG) of the game wherein the game has been played.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

BEST AVAILABLE COPY

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision  
of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-204966

(P2001-204966A)

(43) 公開日 平成13年7月31日 (2001.7.31)

(51) Int.Cl.<sup>7</sup>

A 6 3 F 13/10

識別記号

F I

A 6 3 F 13/10

テーマコード(参考)

2 C 0 0 1

審査請求 未請求 請求項の数 7 O L (全 18 頁)

(21) 出願番号 特願2000-18294(P2000-18294)

(22) 出願日 平成12年1月27日 (2000.1.27)

(71) 出願人 000233778

任天堂株式会社

京都府京都市南区上烏羽鉾立町11番地1

(72) 発明者 飯島 豊

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(72) 発明者 松藤 琢磨

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(74) 代理人 100098291

弁理士 小笠原 史朗

最終頁に続く

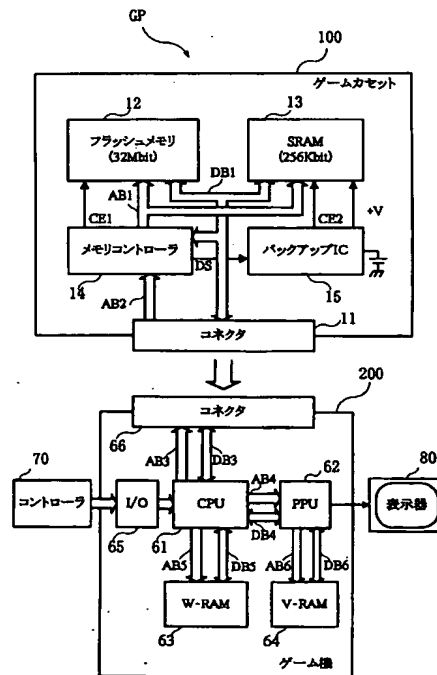
(54) 【発明の名称】 ゲーム装置およびゲーム用情報記憶媒体

(57) 【要約】

【課題】 複数のゲームの進行状態に応じてゲームの展開をより変化させるゲーム装置およびゲームプログラムを記憶した情報記憶媒体を提供する。

【解決手段】 相互に関連する複数のゲームをプレイ可能なゲーム装置 (G P) において、フラッシュメモリ

(12) は相互に関連する複数のゲームプログラム (P G) を記憶し、S R A M (13) は個別のゲーム (A、B、C) の進行状態を表すバックアップデータ (D G) を書き込み読み出し可能に記憶し、プレイヤはコントローラ (70) によって複数のゲームプログラム (P G) のうち一つを選択してゲーム開始と進行を指示し、判定器 (61) はゲーム開始が指示されたときに関連する別のゲームのバックアップデータ (D G) の有無を判定し、ゲームプログラム変更器 (S 2 3 4 ~ S 2 6 2) は判定結果に基づいてゲームプログラム (P G、S) の一部 (a 3 ~ c 7) をゲームプレイ済みのゲームのバックアップデータ (D G) に応じて変更する。



BEST AVAILABLE COPY

## 【特許請求の範囲】

【請求項1】相互に関連する複数のゲームをプレイ可能なゲーム装置であって、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶手段と、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶手段と、前記プログラム記憶手段に記憶された複数のゲームプログラムのうち、いずれか一つを選択してゲーム開始を指示するとともに、ゲームの進行を指示するための操作手段と、前記操作手段によってゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定手段と、前記判定手段の判定結果に基づいて、指示されたゲームプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するゲームプログラム変更手段と、前記変更されたゲームプログラムを読み出す読出手段と、前記読出手段から読み出されたゲームプログラムを実行して画像処理を行うとともに、前記操作手段の操作に基づいて表示画像の変化を行わせるための画像処理を行う画像処理手段とを備えるゲーム装置。

【請求項2】前記バックアップデータ記憶手段は、ゲームをクリアしたとき、その情報を記憶するクリア情報記憶領域を含み、前記判定手段は、指示されたゲームに関連する別のゲームのバックアップデータにクリア情報が書き込まれているか否かを判定することを特徴とする、請求項1に記載のゲーム装置。

【請求項3】前記ゲームプログラム記憶手段に記憶されているゲームプログラムのそれぞれは、複数種類のシナリオユニットから構成されており、前記ゲームプログラム変更手段は、前記判定手段の判定結果に基づいて、前記複数種類のシナリオユニットから予め定められた任意のシナリオユニットを選択してゲームプログラムを構成することを特徴とする、請求項1に記載のゲーム装置。

【請求項4】前記ゲームプログラム記憶手段は、1つのゲームの記憶容量よりも大きなメモリ容量を有し、書き込み読み出し可能な不揮発性メモリであり、前記不揮発性メモリは、相互に関連性を有するように決められた少なくとも2つのゲームのゲームプログラムが予め書込まれており、一方のゲームがプレイされた後他方のゲームがプレイされるように定められ、前記バックアップデータ記憶手段は、前記少なくとも2つのゲームのそれぞれに対応してバックアップデータを

個別に記憶する記憶容量を有することを特徴とする、請求項1に記載のゲーム装置。

【請求項5】前記ゲームプログラム記憶手段は、相互に関連する複数のゲームプログラムとして、ゲームジャンルが同じでありかつゲームに登場する主人公キャラクターが共通し、シナリオの少なくとも一部が異なるゲームのプログラムを記憶することを特徴とする、請求項1に記載のゲーム装置。

【請求項6】操作手段および処理手段を備えたゲーム用情報記憶媒体であって、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶媒体と、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶媒体とを備え、前記ゲームプログラム記憶媒体は、前記操作手段によってゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定プログラムと、前記判定プログラムによる判定結果に基づいて、指示されたゲームのプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するための変更プログラムと、前記変更されたゲームプログラムを読み出す読出制御プログラムと、前記読出制御プログラムにより読み出されたゲームプログラムを実行して画像処理を行わせるとともに、前記操作手段の操作に基づいて表示画像を変化させるための画像処理を行わせるための画像処理プログラムとを含む、ゲーム用情報記憶媒体。

【請求項7】相互に関連する複数のゲームをプレイ可能なゲーム装置を制御するゲームプログラムを記憶した媒体であって、

前記ゲームプログラムは、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶ステップと、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶ステップと、前記複数のゲームプログラムのうち、いずれか一つを選択してゲーム開始を指示するとともに、ゲームの進行を指示するための操作ステップと、ゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定ステップと、前記判定の結果に基づいて、指示されたゲームプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するゲームプログラム変更ステップ

と、  
前記変更されたゲームプログラムを読み出す読出ステップと、  
前記読み出されたゲームプログラムを実行して画像処理を行うとともに、前記操作に基づいて表示画像の変化を行わせるための画像処理を行う画像処理ステップとを前記ゲーム装置に実行させることを特徴とする記憶媒体。

#### 【発明の詳細な説明】

#### 【0001】

【発明の属する技術分野】本発明は、異なる複数のゲームをプレイできるゲーム装置において、それぞれのゲームのシナリオを他のゲームの進行状態に応じて変化させて、同一のゲームであってもゲーム内容に多様性を持たせたゲーム装置とゲームプログラムを記憶した情報記憶媒体に関する。

#### 【0002】

【従来の技術】従来、一つのゲームの中で、そのゲームシナリオ内容に変化を持たせたゲーム機が提供されている。そのようなゲーム機の例として、特許番号第2930237号公報（以下、「先行技術」と称す）に提案されている。この先行技術においては、ゲームにおいて用意された複数の主人公のいずれかを選択する事によって、その主人公に独自に設定されたイベントを実行させることによってゲーム内容に変化を与えている。

#### 【0003】

【発明が解決しようとする課題】上述のように、先行技術においては、ゲーム主人公に独自のイベントを実行することにゲーム内容の多様化を図っている。しかし、ゲームにおいて同一の主人公を選択すれば、ゲームの内容は常に同一である。よって、ゲーム開始当初に感じるゲーム展開の意外性も、ゲーム回数を重ねるに従いイベント実行の意外性がなくなりゲームに対する興味が薄れ、これらのゲームを飽いてしまい易い。

【0004】この発明の主たる目的は、ゲームの展開に複数の異なるゲームのそれぞれにおいて、他のゲームの進行状態に応じてゲームのシナリオ変化させる。結果、複数のゲームの様々な進行状態のそれぞれに応じて、ゲームの展開をより変化に富ますことができ、ゲームの面白さを一層向上する、ゲーム装置およびゲームプログラムを記憶した情報記憶媒体を提供することである。

#### 【0005】

【課題を解決するための手段および発明の効果】第1の発明は、相互に関連する複数のゲームをプレイ可能なゲーム装置であって、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶器と、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶器と、プログラム記憶器に記憶された複数のゲームプログラムのうち、いずれか一つを選択してゲーム開始を指示するととも

に、ゲームの進行を指示するための操作器と、操作器によってゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定器と、判定器の判定結果に基づいて、指示されたゲームプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するゲームプログラム変更器と、変更されたゲームプログラムを読み出す読出器と、読出器から読み出されたゲームプログラムを実行して画像処理を行うとともに、操作器の操作に基づいて表示画像の変化を行わせるための画像処理を行う画像処理器を備える。

【0006】上記のように、第1の発明では、関連する他のゲームの進行状態に応じて、同一のゲームにおいてもその内容を変化させて異なるシナリオを生成できる。

【0007】第2の発明は、第1の発明において、バックアップデータ記憶器は、ゲームをクリアしたとき、その情報を記憶するクリア情報記憶領域を含み、判定器は、指示されたゲームに関連する別のゲームのバックアップデータにクリア情報が書き込まれているか否かを判定することを特徴とする。

【0008】上記のように、第2の発明では、関連する他のゲームのクリア状態に応じて、同一のゲームにおいても内容の異なるシナリオを楽しむことができる。

【0009】第3の発明は、第1の発明において、ゲームプログラム記憶器に記憶されているゲームプログラムのそれぞれは、複数種類のシナリオユニットから構成されており、ゲームプログラム変更器は、判定器の判定結果に基づいて、複数種類のシナリオユニットから予め定められた任意のシナリオユニットを選択してゲームプログラムを構成することを特徴とする。

【0010】上記のように、第3の発明では、シナリオを複数のシナリオユニットで構成し、その複数のシナリオユニットのうちで特定のシナリオユニットに関連する他のゲームのクリア状態に応じて選択することによって、容易に内容の異なるシナリオを構成できる。

【0011】第4の発明は、ゲームプログラム記憶器は、1つのゲームの記憶容量よりも大きなメモリ容量を有し、書込み読み出し可能な不揮発性メモリであり、不揮発性メモリは、相互に関連性を有するように決められた少なくとも2つのゲームのゲームプログラムが予め書込まれており、一方のゲームがプレイされた後他方のゲームがプレイされるように定められ、バックアップデータ記憶器は、少なくとも2つのゲームのそれぞれに対応してバックアップデータを個別に記憶する記憶容量を有することを特徴とする。

【0012】第5の発明は、第1の発明において、ゲームプログラム記憶器は、相互に関連する複数のゲームプログラムとして、ゲームジャンルが同じでありかつゲームに登場する主人公キャラクターが共通し、シナリオの少なくとも一部が異なるゲームのプログラムを記憶するこ

10

20

30

40

50

とを特徴とする。

【0013】第6の発明は、操作器および処理器を備えたゲーム用情報記憶媒体であって、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶媒体と、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶媒体とを備え、ゲームプログラム記憶媒体は、操作器によってゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定プログラムと、判定プログラムによる判定結果に基づいて、指示されたゲームのプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するための変更プログラムと、変更されたゲームプログラムを読み出す読出制御プログラムと、読出制御プログラムにより読み出されたゲームプログラムを実行して画像処理を行わせるとともに、操作器の操作に基づいて表示画像を変化させるための画像処理を行わせるための画像処理プログラムとを含む。

【0014】第7の発明は、相互に関連する複数のゲームをプレイ可能なゲーム装置を制御するゲームプログラムを記憶した媒体であって、ゲームプログラムは、相互に関連する複数のゲームプログラムを記憶するゲームプログラム記憶ステップと、それぞれのゲームを個別にプレイすることによって得られるゲームの進行状態を表すバックアップデータを書込み読み出し可能に記憶するバックアップデータ記憶ステップと、複数のゲームプログラムのうち、いずれか一つを選択してゲーム開始を指示するとともに、ゲームの進行を指示するための操作ステップと、ゲーム開始が指示されたとき、指示されたゲームに関連する別のゲームのバックアップデータがあるか否かを判定する判定ステップと、判定の結果に基づいて、指示されたゲームプログラムの一部をゲームプレイ済みのゲームのバックアップデータに応じて変更するゲームプログラム変更ステップと、変更されたゲームプログラムを読み出す読出ステップと、読み出されたゲームプログラムを実行して画像処理を行うとともに、操作に基づいて表示画像の変化を行わせるための画像処理を行う画像処理ステップとをゲーム装置に実行させることを特徴とする。

【0015】

【発明の実施の形態】図1を参照して、本発明の実施形態にかかるゲーム装置について説明する。本実施形態にかかるゲーム装置GPは、大別してプログラム源100およびゲーム機200を含む。プログラム源100は、ゲーム機200にてゲーム実行に必要なプログラム等の情報を格納し、ゲーム機200に着脱自在に接続できるように構成されている。

【0016】プログラム源100は、好ましくはコネクタ11、フラッシュメモリ12、SRAM13、メモリ

コントローラ14、およびバックアップIC15を含む。プログラム源100は通常、ゲームカセットとしてパッケージ化されている。この意味において、プログラム源100を必要に応じてゲームカセット100と呼称するものとする。フラッシュメモリ12およびSRAM13は、データバスDB1を介してコネクタ11に接続されると共に、アドレスバスAB1を介してメモリコントローラ14に接続される。メモリコントローラ14は、アドレスバスAB2を介してコネクタ11に接続される。

【0017】周知のごとく、フラッシュメモリは、書き換え可能な不揮発性メモリである。本実施形態において、フラッシュメモリ12は、一例として、32Mbitの容量を有しており、後述するように、ゲームプログラムを記憶するために用いられ、4Mbit単位で8つのブロックに分割されている。

【0018】SRAM13は、電源が供給される限り記憶内容を保持する静的なRAMである。本実施形態において、SRAM13は、一例として、256Kbitの容量を有しており、後述するように、ゲームのためのバックアップデータ（攻略したステージ、獲得したアイテム、成長させたキャラクタに関するデータ等）を記憶するために用いられ、16Kbit単位で16個のブロックに分割されている。

【0019】メモリコントローラ14は、アドレスバスAB2を介してカセットライタ20またはゲーム機200から与えられるアドレスを、フラッシュメモリ12およびSRAM13でのメモリ配置に適合するアドレスに変換する機能を有している。バックアップIC15は、図示されていないが、プログラム源100（ゲームカセット）がカセットライタ20またはゲーム機200にセットされているとき、カセットライタ20またはゲーム機200から供給される電力をSRAM13に供給する。

【0020】フラッシュメモリ12およびSRAM13は、メモリコントローラ14から出力されるチップセレクト信号CE1と、バックアップIC15から出力されるチップセレクト信号CE2とによって、それらの選択状態が相補的に切り替えられる。すなわち、フラッシュメモリ12およびSRAM13は、いずれか一方が選択状態になっているとき、いずれか他方が非選択状態になっている。

【0021】これは、フラッシュメモリ12およびSRAM13には、共通のデータバスおよびアドレスバスを介してデータおよびアドレスが供給されるが、供給されたアドレスに対応するメモリチップのアクセスのみを許可状態にするためである。チップセレクト信号CE2は、メモリコントローラ14から出力される1ビットのデコード信号DSおよびSRAM13に対する電力の供給状態を基に、バックアップIC15によって生成され

10

20

30

40

50

る。

【0022】特に、プログラム源100がカセットライタ20またはゲーム機200にセットされた状態であって、これからカセットライタ20またはゲーム機200の電源がオンされて電力が供給されるとき、アドレスバスAB1には、電源オン時の過渡特性によって、ノイズが混入するおそれがある。このノイズが偶然SRAM13のいずれかの番地と一致した場合、メモリコントローラ14が誤ってデコード信号DSを発生しSRAM13の該当する番地には不所望な書き込みが行われる。

【0023】そこで、バックアップIC15は、カセットライタ20またはゲーム機200から供給される電源（電力）の電圧値を監視し、電源オン時には、当該電圧値が予め定める値（約4.5V）に落ち着くまで、チップセレクト信号CE2をSRAM13に与えない。これによって、電源オン時（電力供給時）の不所望な誤書き込みを防止することができる。

【0024】また、メモリコントローラ14からフラッシュメモリ12に与えられるアドレスバスAB1の内、上位3ビットがフラッシュメモリ12の8つに分割されたブロックのいずれかを指定するために用いられる。同様に、メモリコントローラ14からSRAM13に与えられるアドレスバスAB1の内、上位4ビットがSRAM13の16個に分割されたブロックのいずれかを指定するために用いられる。

【0025】ゲーム機200は、CPU61、PPU（ピクチャ・プロセッシング・ユニット）62、ワーキングRAM63、ビデオRAM64、I/Oインターフェイス65、およびコネクタ66を含む。CPU61には、データバスDB3およびアドレスバスAB3を介してコネクタ66が接続され、データバスDB4およびアドレスバスAB4を介してPPU62が接続され、データバスDB5およびアドレスバスAB5を介してワーキングRAM63が接続される。ビデオRAM64は、データバスDB6およびアドレスバスAB6を介してPPU62に接続される。また、PPU62の出力は、テレビジョン受像器などの表示器80に与えられる。

【0026】ゲームを楽しむとき、プログラム源100のコネクタ11がゲーム機200のコネクタ66に差し込まれる。これによって、フラッシュメモリ12およびSRAM13がCPU61の管理下に入る。そして、CPU61は、フラッシュメモリ12からゲームプログラムを読み出し、ゲーム動作を開始する。また、必要に応じて、SRAM13からゲームに対応するバックアップデータを読み出し、ゲーム動作のためのパラメータを設定する。コントローラ70は、プレイヤーによって操作され、CPU61に種々の指示および命令を与える。

【0027】ワーキングRAM63は、CPU61の演算処理に必要な種々のデータを記憶する。PPU62は、CPU61からの命令に基づき、ビデオRAM64

に格納されたキャラクタデータから表示用データを生成する。この表示用データは、表示器80に与えられ、表示される。このように、PPU62を設けることで、CPU61は、表示用データの生成処理から解放され、本来のゲーム動作を高速で行うことが可能となる。

【0028】次に、図2を参照して、フラッシュメモリ12およびSRAM13の記憶領域の構成について詳述する。本実施例におけるフラッシュメモリ12の記憶領域は、大別して、データ記憶領域として、メインプログラム領域APmとゲームプログラム領域APgを含む。メインプログラム領域APmには、ゲーム選択プログラム、判定プログラム、変更プログラム、および読出・書込制御プログラムが記憶される。

【0029】ゲームプログラム領域APgには、それぞれ異なるゲームを実行するゲームAプログラムPGa、ゲームBプログラムPGb、およびゲームCプログラムPGcが記憶される。なお、本例においては、ゲームA、ゲームB、およびゲームCの3種類（数）のゲームプログラムが記憶されているが、フラッシュメモリ12の容量の許す限り任意の種類（数）のゲームプログラムを記憶しても良いことは言うまでもない。

【0030】ゲームAプログラムPGaには、スタート設定プログラム、画像処理プログラム、ゲームタイトル（バージョン）、シナリオSAを構成するシナリオユニットa1、a2、a3～a7、およびa8、およびキャラクタデータのゲームAの実行に必要なプログラムデータが含まれる。同様に、ゲームBプログラムPGbには、ゲームBの実行に必要なプログラムデータであるスタート設定プログラム、画像処理プログラム、ゲームタイトル（バージョン）、シナリオSBを構成するシナリオユニットb1、b2、b3～b7、およびb8、およびキャラクタデータが含まれる。

【0031】さらに、ゲームCプログラムPGcには、ゲームCの実行に必要なプログラムデータであるスタート設定プログラム、画像処理プログラム、ゲームタイトル（バージョン）、シナリオSCを構成するシナリオユニットc1、c2、c3～c7、およびc8、およびキャラクタデータが含まれる。なお、これらのゲームA、ゲームB、およびゲームCは、ゲームジャンルが同じでありかつゲームに登場する主人公キャラクタが共通し、シナリオの少なくとも一部が異なるという点において、相互に関連するゲームと言える。

【0032】一方、SRAM13には、上述のフラッシュメモリ12のゲームプログラム領域APgに書き込まれたゲームプログラムPGa、PGb、およびPGcのそれぞれのためのバックアップデータが記憶される。例えば、ゲームAバックアップデータDGaには、ゲームAプログラムPGaのゲームタイトル（バージョン）DTa、プレイヤーの識別コード、ゲームの進行状態を表す途中経過データDPa、およびクリアデータDCa等が

含まれる。同様に、ゲームBバックアップデータD G bおよびゲームCバックアップデータD G cには、図2に示されるように、それぞれゲームBプログラムP G bおよびゲームCプログラムP G cのバックアップデータが含まれる。

【0033】例えば、ゲームAを実行中には、S R A M 1 3のゲームAバックアップデータD G aにゲームタイトルD T aが自動的に記録される。また、プレイヤーがゲームAをクリアした時点で初めて、クリアデータD C aが自動的に記録される。クリアデータD C aには、その時点で他の関連するゲームのクリアデータD C bおよびD C cの有無に基づいて、クリア順序が記録される。

【0034】一方、途中経過データD P aは、プレイヤーがゲームを中断（終了）する際に、プレイ中のゲームの状態を保存すると言う行為をもって初めて記録される。なお、前述のゲームタイトルD T aは、経過データD P aと同時に記録されるようにしても良い。また、上述の如く本実施形態においては、メインプログラム（A P）およびゲームプログラムP Gをフラッシュメモリ12に記憶し、バックアップデータD GをS R A M 1 3に記憶するようにしたが、これらを1つのメモリ内に記憶させるようにしても良い。

【0035】先ず、図3に示すシナリオユニット選択条件表を参照して、本実施形態において、プレイヤーが各ゲームを開始した際に、そのゲームを開始した時の状況に応じて異なる内容を有するシナリオが生成されることについて説明する。まず、ゲームAは複数のシナリオユニットを選択的に組み合わせる構成される。

【0036】本例においては、ゲームAのシナリオS Aは、a 1、a 2、a 3、a 4、a 5、a 6、a 7、およびa 8の8つのシナリオユニットを選択的に組み合わせることによって構成されるが、必要に応じて、さらにフラッシュメモリ12の容量の制限に応じて8つ以外の適当な数のシナリオユニットから構成するようにしても良い。

【0037】同図に示されるように、シナリオユニットa 1およびa 2は、ゲームAをプレイする場合には、シナリオS Aを構成するために無条件に常に選択される。

【0038】シナリオユニットa 3は、ゲームAが初めてプレイされる時にのみ選択される。

【0039】シナリオユニットa 4は、既にゲームBのみがクリアされた状態で、ゲームAをプレイする時に選択される。

【0040】シナリオユニットa 5は、既にゲームCのみがクリアされた状態で、ゲームAをプレイする時に選択される。

【0041】シナリオユニットa 6は、既にゲームB、ゲームCの順番に両ゲームがクリアされた状態で、ゲームAをプレイする時に選択される。

【0042】シナリオユニットa 7は、既にゲームC、

ゲームBの順番に両ゲームがクリアされた状態で、ゲームAをプレイする時に選択される。

【0043】シナリオユニットa 8は、ゲームAをプレイする場合には無条件に、常に選択される。

【0044】図4を参照して、ゲームAの場合を例にして、上述の条件に従ってシナリオユニットのそれぞれを選択的に組み合わせることによって、異なるストーリーを有するように発生されるゲームシナリオS Aについて説明する。

10 【0045】同図に示すように、図3を参照して説明した条件に従って、初めてゲームAをプレイする場合には、シナリオユニットa 1、a 2、a 3、およびa 8から構成されるシナリオS A αが生成される。

【0046】既にゲームBのみがクリアされた状態でゲームAをプレイする時には、シナリオユニットa 1、a 2、a 4、およびa 8から構成されるシナリオS A βが生成される。

20 【0047】既にゲームCのみがクリアされた状態でゲームAをプレイする時には、シナリオユニットa 1、a 2、a 5、およびa 8から構成されるシナリオS A γが生成される。

【0048】既にゲームB、ゲームCの順番に両ゲームがクリアされた状態でゲームAをプレイする時には、シナリオユニットa 1、a 2、a 6、およびa 8から構成されるシナリオS A δが生成される。

【0049】既にゲームC、ゲームBの順番に両ゲームがクリアされた状態で、ゲームAをプレイする時には、シナリオユニットa 1、a 2、a 7、およびa 8から構成されるシナリオS A εが生成される。

30 【0050】このように、本例においては、ゲームAは4つのシナリオユニットから異なる内容のストーリーを有する5つのシナリオS A α、S A β、S A γ、S A δ、およびS A εが構成される。これら5つのシナリオS A α、S A β、S A γ、S A δ、およびS A εのいずれにおいても、4つのシナリオユニットの先頭の2つと、最後の1つは、それぞれシナリオユニットa 1、a 2、およびa 8で固定されている。

【0051】つまり、図3に示した条件によって、第3番目のシナリオユニットとして、5種類のシナリオユニットa 3、a 4、a 5、a 6、およびa 7から一つがゲーム開始時の状況に応じて選択される。なお、このように、第1番目、第2番目、および第3番目のシナリオユニットを固定するばかりでなく、全シナリオユニットa 1～a 8から選択するように条件設定しても良く、第3番目のシナリオユニットを固定しても良い。また、ゲームシナリオS Aは、4つ以外の数のシナリオユニットを組み合わせても良いし、さらにシナリオユニットのそれぞれを固定あるいは条件で選択するようにしても良いことは言うまでもない。

50 【0052】図3および図4に明示されているように、



ゲームBおよびゲームCについても、上述したゲームAと同様に、それぞれのシナリオユニットがゲーム開始時の状況に応じて選択されて、それぞれ異なるシナリオS BおよびシナリオS Cが生成される。

【0053】つまり、ゲームBに関して、シナリオユニットb 1およびb 2は、ゲームBをプレイする場合には無条件に、常に選択される。

【0054】シナリオユニットb 3は、ゲームBが初めてプレイされる時にのみ選択される。

【0055】シナリオユニットb 4は、既にゲームAのみがクリアされた状態で、ゲームBをプレイする時に選択される。

【0056】シナリオユニットb 5は、既にゲームCのみがクリアされた状態で、ゲームBをプレイする時に選択される。

【0057】シナリオユニットb 6は、既にゲームA、ゲームCの順番に両ゲームがクリアされた状態で、ゲームBをプレイする時に選択される。

【0058】シナリオユニットb 7は、既にゲームC、ゲームAの順番に両ゲームがクリアされた状態で、ゲームBをプレイする時に選択される。

【0059】シナリオユニットb 8は、ゲームBをプレイする場合には無条件に、常に選択される。

【0060】よって、初めてゲームBをプレイする場合には、シナリオユニットb 1、b 2、b 3、およびb 8から構成されるシナリオS B  $\alpha$ が生成される。

【0061】既にゲームAのみがクリアされた状態でゲームBをプレイする時には、シナリオユニットb 1、b 2、b 4、およびb 8から構成されるシナリオS B  $\beta$ が生成される。

【0062】既にゲームCのみがクリアされた状態でゲームBをプレイする時には、シナリオユニットb 1、b 2、b 5、およびb 8から構成されるシナリオS B  $\gamma$ が生成される。

【0063】既にゲームA、ゲームCの順番に両ゲームがクリアされた状態でゲームBをプレイする時には、シナリオユニットb 1、b 2、b 6、およびb 8から構成されるシナリオS B  $\delta$ が生成される。

【0064】既にゲームC、ゲームAの順番に両ゲームがクリアされた状態で、ゲームBをプレイする時には、シナリオユニットb 1、b 2、b 7、およびb 8から構成されるシナリオS B  $\epsilon$ が生成される。

【0065】そして、ゲームCに関して、シナリオユニットc 1およびc 2は、ゲームCをプレイする場合には無条件に、常に選択される。

【0066】シナリオユニットc 3は、ゲームCが初めてプレイされる時にのみ選択される。

【0067】シナリオユニットc 4は、既にゲームAのみがクリアされた状態で、ゲームCをプレイする時に選択される。

【0068】シナリオユニットc 5は、既にゲームBのみがクリアされた状態で、ゲームCをプレイする時に選択される。

【0069】シナリオユニットc 6は、既にゲームA、ゲームBの順番に両ゲームがクリアされた状態で、ゲームCをプレイする時に選択される。

【0070】シナリオユニットc 7は、既にゲームB、ゲームAの順番に両ゲームがクリアされた状態で、ゲームCをプレイする時に選択される。

【0071】シナリオユニットc 8は、ゲームCをプレイする場合には無条件に、常に選択される。

【0072】結果、初めてゲームCをプレイする場合には、シナリオユニットc 1、c 2、c 3、およびc 8から構成されるシナリオS C  $\alpha$ が生成される。

【0073】既にゲームAのみがクリアされた状態でゲームCをプレイする時には、シナリオユニットc 1、c 2、c 4、およびc 8から構成されるシナリオS C  $\beta$ が生成される。

【0074】既にゲームAのみがクリアされた状態でゲームCをプレイする時には、シナリオユニットc 1、c 2、c 5、およびc 8から構成されるシナリオS C  $\gamma$ が生成される。

【0075】既にゲームA、ゲームBの順番に両ゲームがクリアされた状態でゲームCをプレイする時には、シナリオユニットc 1、c 2、c 6、およびc 8から構成されるシナリオS C  $\delta$ が生成される。

【0076】既にゲームB、ゲームAの順番に両ゲームがクリアされた状態で、ゲームCをプレイする時には、シナリオユニットc 1、c 2、c 7、およびc 8から構成されるシナリオS C  $\epsilon$ が生成される。

【0077】ゲームBおよびゲームCのシナリオ生成に関しても、ゲームAのシナリオ生成と同様に、4つのシナリオユニットの先頭の2つと、最後の1つは、それぞれ固定されているが、条件によって全シナリオユニットから選択するようにしてもよし、第3番目のシナリオユニットを固定しても良い。また、ゲームシナリオS BおよびS Cは、4つ以外の数のシナリオユニットを組み合わせても良いし、さらにシナリオユニットのそれぞれを固定あるいは条件で選択するようにしても良いことは言うまでもない。

【0078】なお、図4の右端部には、フラグ欄が設けられている。このフラグ欄には、F 1～F 15（順不同）が明記されている。これらのフラグF 1～F 15のそれぞれは、同図において記されている「発生条件」に対応している。これらフラグF 1～F 15に関しては、後ほど図6、図7、図8、図9、および図10を参照して詳しく説明する。

【0079】次に、図5に示すフローチャートを参照して、本発明の実施形態にかかるゲーム装置G Pによるシナリオ生成動作について説明する。ゲーム装置G Pの電

源がONにされてその動作が開始すると、先ずステップS2において、表示器80にゲームA、ゲームB、およびゲームCに関するメニュー画面が表示される。そして、処理は次のステップS4に進む。

【0080】ステップS4において、プレイヤによるコントローラ70の動作に基づいて、メニュー画面に表示されているゲームA、ゲームB、およびゲームCのいずれが選択されたかが判断(判定)される。プレイヤがいずれのゲームも選択していない場合はNoと判断されて、処理は上述のステップS2に戻る。一方、ゲームA、ゲームB、およびゲームCのいずれかが選択された場合にはYesと判断されて、処理は次のステップS6に進む。

【0081】ステップS6において、ステップS4において選択されたと判断されたゲームのタイトルあるいはデモ画面が表示器80に表示される。そして、処理は次のステップS8に進む。

【0082】ステップS8において、ステップS4で選択されたゲームを開始するために、プレイヤがコントローラ70に設けられたスタートボタンをONにしたのか否か判断される。Noの場合は、処理はステップS6に戻って、タイトルあるいはデモ画面の表示を継続して、プレイヤによるゲーム開始指示の入力を待つ。一方、Yesの場合は、処理は次のステップS10に進む。

【0083】ステップS10において、ステップS4において選択が判断され、ステップS8において開始が判断されたゲームが、かつて一度もプレイされたことがなく、初めてプレイされるのか否かが判断される。この判断は、SRAM13に記録されたゲームバックアップデータDGに基づいて成される。

【0084】つまり、選択されたゲームに対応するクリアデータDC、経過データDP、ゲームタイトルDTのいずれかが記録されていれば、このゲームは初めてのプレイではないと判断される。

【0085】Yesの場合、処理はステップ#100のスタート時チェックサブルーチンの処理を経て、ステップS16に進む。なお、ステップ#100のスタート時チェックサブルーチンにおける処理の内容については、後ほど図6、図7、および図8を参照して詳しく述べる。

【0086】一方、ステップS10においてNo、つまり今回開始するゲームは初めてのプレイでない場合には、処理はステップS12に進む。

【0087】ステップS12においては、SRAM13に格納されているゲームAバックアップデータDGa、ゲームBバックアップデータDGb、およびゲームCバックアップデータDGcのうち、選択されたゲームに対応するものを読み出す。そして、処理は次のステップS14に進む。

【0088】ステップS14において、読み出したゲー

ムAバックアップデータDGa、ゲームBバックアップデータDGb、またはゲームCバックアップデータDGcの内容に基づいて、ゲーム開始時の設定が行われる。そして、処理はステップS16に進む。

【0089】ステップ#100あるいはステップS14における処理において、選択されたゲームに関する種々の設定が成された後に、ステップS16において選択されたゲームのゲームプログラムPGが実行される。そして、所定量だけゲームを実行した後、処理は次のステップS18に進む。

【0090】ステップS18において、コントローラ70の操作に基づいて、プレイヤがゲームの終了を指示しているか否かが判断される。Noの場合は、処理はステップS16に戻って、また所定量だけゲーム実行を継続する。一方、Yesの場合は、処理は次のステップS20に進む。

【0091】ステップS20において、コントローラ70の操作およびゲームプログラムの実行状態に基づいて、プレイヤはゲームをクリアしたのか否かが判断される。Yesの場合は、処理はステップS22に進む。

【0092】そして、ステップS22において、クリアデータSRAM13のバックアップデータDGのクリアデータ領域DCにクリアデータが書き込まれる。そして、処理は終了する。

【0093】一方、ステップS20においてNo、つまりプレイヤは実行中のゲームをまだクリアしていない場合には、処理はステップS24に進む。

【0094】ステップS24においては、コントローラ70の操作およびゲームプログラムの実行状態に基づいて、プレイヤはゲームを終了(中断)する前に、ゲームのプレイ(進行)状態の保存を指示しているか否かが判断される。Yesの場合は、処理はステップS26に進む。

【0095】ステップS26において、SRAM13のバックアップデータDGの途中経過データ領域DPに途中経過データが書き込まれる。そして、処理を終了する。また、ステップS24においてNo、つまりプレイヤはゲームをクリアせずに、途中経過を保存することも指示していない場合には、そのまま処理を終了する。

【0096】次に、図6を参照して、上述のステップ#100のスタート時チェックのサブルーチンにおける処理について説明する。

【0097】前述のステップS10において、選択されたゲームが初めてプレイされると判断される場合には、先ず、ステップS102において、選択されたゲームのタイトルがフラッシュメモリ12のゲームプログラム領域APgから読み出される。そして処理は、次のステップS104に進む。

【0098】ステップS104において、図3および図4を参照して説明したように、選択されたゲームのシナ

リオを生成するために用いられるシナリオユニットの選択に影響を与える他のゲームのバックアップデータDGがSRAM13に記憶されているか否かが判断される。

【0099】本例においては、バックアップデータDGの中でも特にクリアデータDCに基づいて、シナリオユニットの選択が決定される。つまり、Yesの場合には、処理は図7に示すステップS106～S128および図8に示すステップS130～S152の処理を経た後に、図6に示すステップS154およびステップ#200のゲームプログラム設定サブルーチンに進む。

【0100】上述のステップS104でYesと判断された場合には、図7に示すステップS106において、SRAM13に記憶されているクリアデータDCはゲームAのクリアデータDCaのみか否かが判断される。Yes、つまりゲームAのみが既にクリアされている場合は、処理は次のステップS108に進む。

【0101】ステップS108において、先のステップS102において読み出されたゲームのタイトルがゲームBのタイトルか否かが判断される。Yes、つまり選択されたゲームがゲームBの場合には、処理は次のステップS110に進む。

【0102】ステップS110において、フラグF4がセットされる。そして、処理は図6に示すステップ#200に進む。つまり、フラグF4は、ゲームBを開始するに当たって、ゲームAだけがクリアされていることを示す。

【0103】一方、ステップS108でNo、つまりゲームAのみがクリアされていて、選択されたゲームがゲームCの場合には、処理はステップS112に進む。

【0104】ステップS112において、フラグF5が30 セットされる。そして、処理は図6に示すステップ#200に進む。つまり、フラグF5は、ゲームCを開始するに当たって、ゲームAだけがクリアされていることを示す。

【0105】上述のステップS106でNo、つまりゲームAだけがクリアされているのではない場合には、処理はステップS114に進む。

【0106】ステップS114において、SRAM13に記憶されているクリアデータDCはゲームBのクリアデータDCbのみか否かが判断される。Yes、つまりゲームBのみが既にクリアされている場合は、処理は次のステップS116に進む。

【0107】ステップS116において、上述のステップS102において読み出されたゲームのタイトルがゲームAのタイトルか否かが判断される。Yes、つまり選択されたゲームがゲームAの場合には、処理は次のステップS118に進む。

【0108】ステップS118において、フラグF6が40 セットされる。そして、処理は図6に示すステップ#200に進む。フラグF6は、ゲームAを開始するに当た

って、ゲームBだけがクリアされていることを示す。

【0109】一方、ステップS116においてNo、つまりゲームBのみがクリアされていて、選択されたゲームがゲームCの場合には、処理はステップS120に進む。

【0110】ステップS120において、フラグF7がセットされる。そして、処理は図6に示すステップ#200に進む。フラグF7は、ゲームCを開始するに当たって、ゲームBだけがクリアされていることを示す。

10 【0111】上述のステップS114でNo、つまりゲームBだけがクリアされているのではない場合には、処理はステップS122に進む。

【0112】ステップS122において、SRAM13に記憶されているクリアデータDCはゲームCのクリアデータDCcのみか否かが判断される。Yes、つまりゲームCのみが既にクリアされている場合は、処理は次のステップS124に進む。

20 【0113】ステップS124において、上述のステップS102において読み出されたゲームのタイトルがゲームAのタイトルか否かが判断される。Yes、つまり選択されたゲームがゲームAの場合には、処理は次のステップS126に進む。

【0114】ステップS126において、フラグF8がセットされる。そして処理は図6に示すステップ#200に進む。つまり、フラグF8は、ゲームAを開始するに当たって、ゲームCだけがクリアされていることを示す。

【0115】一方、ステップS124においてNo、つまりゲームCのみがクリアされていて、選択されたゲームがゲームBの場合には、処理はステップS128に進む。

【0116】ステップS128において、フラグF9がセットされる。そして、処理はステップ#200に進む。つまり、フラグF9は、ゲームBを開始するに当たって、ゲームCだけがクリアされていることを示す。

【0117】上述のステップS122でNo、つまりゲームCだけがクリアされているのではない場合には、処理は図8に示すステップS130に進む。

40 【0118】ステップS130において、SRAM13にクリアデータDCaとクリアデータDCbが記憶されているかが判断される。Yes、つまりゲームAとゲームBが既にクリアされている場合は、処理は次のステップS132に進む。

【0119】ステップS132において、クリアデータDCaが先か否かが判断される。Yes、つまりゲームAはゲームBより先にクリアされている場合には、処理は次のステップS134に進む。

【0120】ステップS134において、フラグF10が50 セットされる。そして、処理はステップ#200に進む。つまり、フラグF10は、ゲームCを開始するに当

たって、既にゲームA、ゲームBの順番にクリアされていることを示す。

【0121】一方、ステップS132においてNo、つまりゲームBの方がゲームAより先にクリアされている場合には、処理はステップS136に進む。

【0122】ステップS136において、フラグF11がセットされる。そして、処理はステップ#200に進む。つまり、フラグF11は、ゲームCを開始するに当たって、既にゲームB、ゲームAの順番にクリアされていることを示す。

【0123】上述のステップS130でNo、つまりゲームAとゲームBの両方がクリアされているのではない場合には、ステップS138に進む。

【0124】ステップS138において、SRAM13にクリアデータDCaとクリアデータDCcが記憶されているかが判断される。Yes、つまりゲームAとゲームCが既にクリアされている場合は、処理は次のステップS140に進む。

【0125】ステップS140において、クリアデータDCaが先か否かが判断される。Yes、つまりゲームAはゲームCより先にクリアされている場合には、処理は次のステップS142に進む。

【0126】ステップS142において、フラグF12がセットされる。そして、処理はステップ#200に進む。つまり、フラグF12は、ゲームBを開始するに当たって、既にゲームA、ゲームCの順番にクリアされていることを示す。

【0127】一方、ステップS140においてNo、つまりゲームCの方がゲームAより先にクリアされている場合には、処理はステップS144に進む。

【0128】ステップS144において、フラグF13がセットされる。そして、処理はステップ#200に進む。つまり、フラグF13は、ゲームBを開始するに当たって、既にゲームC、ゲームAの順番にクリアされていることを示す。

【0129】上述のステップS138においてNo、つまりゲームAとゲームCの両方がクリアされているのではない場合には、処理はステップS146に進む。

【0130】ステップS146において、SRAM13にクリアデータDCbとクリアデータDCcが記憶されているかが判断される。Yes、つまりゲームBとゲームCが既にクリアされている場合は、処理は次のステップS148に進む。

【0131】ステップS148において、クリアデータDCbが先か否かが判断される。Yes、つまりゲームBはゲームCより先にクリアされている場合には、処理は次のステップS150に進む。

【0132】ステップS150において、フラグF14がセットされる。そして、処理はステップ#200に進む。つまり、フラグF14は、ゲームAを開始するに当

たって、既にゲームB、ゲームCの順番にクリアされていることを示す。

【0133】一方、ステップS148においてNo、つまりゲームCの方がゲームBより先にクリアされている場合には、処理はステップS152に進む。

【0134】ステップS152において、フラグF15がセットされる。そして、処理はステップ#200に進む。つまり、フラグF15は、ゲームAを開始するに当たって、既にゲームC、ゲームBの順番にクリアされていることを示す。

【0135】上述のステップS146でNo、つまりゲームBとゲームCの両方がクリアされているのではない場合には、処理は図6に示すステップS154に進む。このようにして、図3および図4を参照して説明した、シナリオユニット選択の条件のそれぞれに対応するフラグF4～フラグF15がセットされる。

【0136】図6に示したステップS104において、クリアデータDCがあると判断された場合は、図7および図8に示した処理を経て、ゲームA、ゲームB、およびゲームCの内一つだけ、あるいはその2つがクリアされていない場合、すなわち、クリアされたゲームがない場合には、シナリオユニット選択条件に対応するフラグをセットされることなく、処理は同図に示すステップS154に進む。

【0137】また、ステップS104でクリアデータDCがないと判断された場合、シナリオユニット選択条件に対応するフラグをセットすることなく、処理はステップS154に進む。つまり、ステップS154の処理が開始される時には、フラグF4～F15は一切セットされていない。

【0138】ステップS154において、上述のステップS102において読み出されたゲームのタイトルがゲームAのタイトルか否かが判断される。Yes、つまり選択されたゲームがゲームAの場合には、フラグF1がセットされる。そして処理はステップ#200のゲームプログラム設定サブルーチンに進む。つまり、フラグF1は、今回初めてゲームAがプレイされることを示す。

【0139】一方、ステップS154においてNo、つまり選択されたゲームがゲームAでない場合は、処理はステップS158に進む。

【0140】ステップS158において、上述のステップS102において読み出されたゲームのタイトルがゲームBのタイトルか否かが判断される。Yes、つまり選択されたゲームがゲームBの場合には、フラグF2がセットされる。そして処理はステップ#200に進む。つまり、フラグF2は、今回初めてゲームBがプレイされることを示す。

【0141】一方、ステップS158においてNo、つまり選択されたゲームがゲームBでない場合は、処理はステップS162に進む。

【0142】ステップS162において、選択されたゲームはゲームCであることから、フラグF3がセットされる。そして、処理はステップ#200に進む。つまり、フラグF3は、今回初めてゲームCがプレイされることを示す。

【0143】次に、図9および図10を参照して、ステップ#200のゲームプログラム設定サブルーチンにおける処理について詳細に説明する。上述の処理を経て、シナリオユニット選択条件をフラグF1～F15として検出された状態で、ステップ#200の処理が開始される。なお、フラグF1～F15と検出されたシナリオユニット選択条件の対応関係が上述の図4に示されている。

【0144】まず、ステップS202において、各ゲームのシナリオSA、SB、およびSCのそれぞれを構成する第1番目のシナリオユニットa1、b1、およびc1から、上述のステップS4において選択が認識されたゲームに対応するものが選択される。そして、処理は次のステップS204に進む。

【0145】ステップS204においては、各ゲームのシナリオSA、SB、およびSCのそれぞれを構成する第2番目のシナリオユニットa2、b2、およびc2から、ステップS202で選択された1番目のシナリオユニットに対応するものが選択される。そして、処理は次のステップS206に進む。

【0146】ステップS206において、フラグF1がセットされているか否かが判断される。フラグF1がセットされている、つまり今回初めてゲームAがプレイされる場合には、処理はステップS208に進む。

【0147】ステップS208において、シナリオSA $\alpha$ を構成する第3番目のシナリオユニットとしてシナリオユニットa3が選択される。そして処理はステップS262に進む。一方、ステップS206においてNoと判断される場合は、処理はステップS210に進む。

【0148】ステップS210において、フラグF2がセットされているか否かが判断される。フラグF2がセットされている、つまり今回初めてゲームBがプレイされる場合には、処理はステップS212に進む。

【0149】ステップS212において、シナリオSB $\alpha$ を構成する第3番目のシナリオユニットとしてシナリオユニットb3が選択される。そして処理はステップS262に進む。一方、ステップS210においてNoと判断される場合は、処理はステップS214に進む。

【0150】ステップS214において、フラグF3がセットされているか否かが判断される。フラグF3がセットされている、つまり今回初めてゲームCがプレイされる場合には、処理はステップS216に進む。

【0151】ステップS216において、シナリオSC $\alpha$ を構成する第3番目のシナリオユニットとしてシナリオユニットc3が選択される。そして処理はステップS

262に進む。一方、ステップS214においてNoと判断される場合は、処理はステップS218に進む。

【0152】ステップS218において、フラグF4がセットされているか否かが判断される。フラグF4がセットされている、つまりゲームBを開始するに当たって、ゲームAだけがクリアされている場合には、処理はステップS220に進む。

【0153】ステップS220において、シナリオSB $\beta$ を構成する第3番目のシナリオユニットとしてシナリオユニットb4が選択される。そして処理はステップS262に進む。一方、ステップS218においてNoと判断される場合は、処理はステップS222に進む。

【0154】ステップS222において、フラグF5がセットされているか否かが判断される。フラグF5がセットされている、つまりゲームCを開始するに当たって、ゲームAだけがクリアされている場合には、処理はステップS224に進む。

【0155】ステップS224において、シナリオSC $\beta$ を構成する第3番目のシナリオユニットとしてシナリオユニットc4が選択される。そして処理はステップS262に進む。一方、ステップS222においてNoと判断される場合は、処理はステップS226に進む。

【0156】ステップS226において、フラグF6がセットされているか否かが判断される。フラグF6がセットされている、つまりゲームAを開始するに当たって、ゲームBだけがクリアされている場合には、処理はステップS228に進む。

【0157】ステップS228において、シナリオSA $\beta$ を構成する第3番目のシナリオユニットとしてシナリオユニットa4が選択される。そして処理はステップS262に進む。一方、ステップS226においてNoと判断される場合は、処理はステップS230に進む。

【0158】ステップS230において、フラグF7がセットされているか否かが判断される。フラグF7がセットされている、つまりゲームCを開始するに当たって、ゲームBだけがクリアされている場合には、処理はステップS232に進む。

【0159】ステップS232において、シナリオSC $\gamma$ を構成する第3番目のシナリオユニットとしてシナリオユニットc5が選択される。そして処理はステップS262に進む。一方、ステップS230においてNoと判断される場合は、処理は図10に示すステップS234に進む。

【0160】ステップS234において、フラグF8がセットされているか否かが判断される。フラグF8がセットされている、つまりゲームAを開始するに当たって、ゲームCだけがクリアされている場合には、処理はステップS236に進む。ステップS236において、シナリオSA $\gamma$ を構成する第3番目のシナリオユニットとしてシナリオユニットa5が選択される。そして処理

は図9に示したステップS262に進む。一方、ステップS234においてNoと判断される場合は、処理はステップS238に進む。

【0161】ステップS238において、フラグF9がセットされているか否かが判断される。フラグF9がセットされている、つまりゲームBを開始するに当たって、ゲームCだけがクリアされている場合には、処理はステップS240に進む。ステップS240において、シナリオS $\beta$ を構成する第3番目のシナリオユニットとしてシナリオユニットb5が選択される。そして処理はステップS262に進む。一方、ステップS238においてNoと判断される場合は、処理はステップS242に進む。

【0162】ステップS242において、フラグF10がセットされているか否かが判断される。フラグF10がセットされている、つまりゲームCを開始するに当たって、既にゲームA、ゲームBの順番にクリアされている場合には、処理はステップS244に進む。

【0163】ステップS244において、シナリオSC $\delta$ を構成する第3番目のシナリオユニットとしてシナリオユニットc6が選択される。そして処理はステップS262に進む。一方、ステップS242においてNoと判断される場合は、処理はステップS246に進む。

【0164】ステップS246において、フラグF11がセットされているか否かが判断される。フラグF11がセットされている、つまりゲームCを開始するに当たって、既にゲームB、ゲームAの順番にクリアされている場合には、処理はステップS248に進む。

【0165】ステップS248において、シナリオSC $\epsilon$ を構成する第3番目のシナリオユニットとしてシナリオユニットc7が選択される。そして処理はステップS262に進む。一方、ステップS246においてNoと判断される場合は、処理はステップS250に進む。

【0166】ステップS250において、フラグF12がセットされているか否かが判断される。フラグF12がセットされている、つまりゲームBを開始するに当たって、既にゲームA、ゲームCの順番にクリアされている場合には、処理はステップS252に進む。

【0167】ステップS252において、シナリオSB $\delta$ を構成する第3番目のシナリオユニットとしてシナリオユニットb6が選択される。そして処理はステップS262に進む。一方、ステップS250においてNoと判断される場合は、処理はステップS254に進む。

【0168】ステップS254において、フラグF13がセットされているか否かが判断される。フラグF13がセットされている、つまりゲームBを開始するに当たって、既にゲームC、ゲームAの順番にクリアされている場合には、処理はステップS256に進む。

【0169】ステップS256において、シナリオSB $\epsilon$ を構成する第3番目のシナリオユニットとしてシナリ

オユニットb7が選択される。そして処理はステップS262に進む。一方、ステップS254においてNoと判断される場合は、処理はステップS258に進む。

【0170】ステップS258において、フラグF14がセットされているか否かが判断される。フラグF14がセットされている、つまりゲームAを開始するに当たって、既にゲームB、ゲームCの順番にクリアされている場合には、処理はステップS260に進む。

【0171】ステップS260において、シナリオSA $\delta$ を構成する第3番目のシナリオユニットとしてシナリオユニットa6が選択される。そして処理はステップS262に進む。

【0172】一方、ステップS258においてNoと判断される場合は、フラグF15がセットされている。つまり、ゲームAを開始するに当たって、既にゲームC、ゲームBの順番にクリアされている。それゆえ、シナリオSA $\epsilon$ を構成する第3番目のシナリオユニットとしてシナリオユニットa7が選択される。そして処理はステップS262に進む。

【0173】ステップS262において、各ゲームのシナリオSA、SB、およびSCのそれぞれを構成する第4番目のシナリオユニットa8、b8、およびc8から、上述のステップS4において選択が認識されたゲームに対応するものが選択される。

【0174】上述のように、図6、図7、および図8を参照して説明したステップ#100のスタート時チェックサブルーチンにおいて、互いに関連づけられたゲームA、ゲームB、およびゲームCの相互関係を検出し、検出した相互関係をフラグF1～F15に代表させ、これらのフラグF1～F15のセット状態に基づいて、各ゲームシナリオSA、SB、およびSCを構成するシナリオユニットの組み合わせを変えることによって、例えば、同一のゲームAであってもSA $\alpha$ 、SA $\beta$ 、SA $\gamma$ 、SA $\delta$ 、およびSA $\epsilon$ という五種類の内容を有するシナリオを生成して、ゲームに多様性を持たせることができる。なお、ゲームBおよびゲームCに関しても同様であることは、上記より明白である。

【0175】また、本発明の実施形態においては、シナリオユニット選択条件として、関連するゲームA、ゲームB、およびゲームCの単独での進行状況、あるいはそれぞれの進行状況を他のゲームの進行状況との相互関係に対応づけた方法が典型的な例として図3および図4に示されている。しかしながら、このようなシナリオユニットの選択条件は、例えば、一つのシナリオを構成する各シナリオユニットのクリア時間に基づいて、後続のシナリオユニットを選択するようにしても良い。

【0176】また、所定の時間あるいは一つのシナリオユニットをクリアする間に、プレイヤーが操作した特定あるいは不定のキーのストローク数に基づいて、後続のシナリオユニットを選択するようにしても良い。また、ゲ

ーム装置G Pの内部の時計によって計時される時間情報に基づいて、後続のシナリオユニットを選択することも考えられる。このように、プレイヤの操作あるいは非操作にかかわらず変化する量に基づいて、シナリオユニットを選択するようにすることで、同一のゲームの展開に多様性を持たすことができる。なお、このような変化量としては、プレイヤが意識し難い量を設定することによって、より一層意外性に富んだゲームの展開を実現できる。

【図面の簡単な説明】

【図1】本発明にかかるゲーム装置の構成を示すブロック図である。

【図2】図1に示したフラッシュメモリおよびSRAMの記憶領域の構成を示す模式図である。

【図3】図1に示したゲーム装置において、関連するゲーム間の相互関係に基づくシナリオユニット選択条件を示す表である。

【図4】図4に示したシナリオユニット選択条件にもとづいて構成されるシナリオとシナリオユニットの関係を示す表である。

【図5】本発明にかかるゲーム装置におけるシナリオ生成の動作を示すフローチャートである。

【図6】図5に示したステップ#100における処理の一部を示すフローチャートである。

【図7】図5に示したステップ#100における処理の一部であって、図6に示したのとは別の部分の動作を示すフローチャートである。

【図8】図5に示したステップ#100における処理の

一部であって、図6および図7に示したのとは別の部分の動作を示すフローチャートである。

【図9】図6に示したステップ#200における処理の一部を示すフローチャートである。

【図10】図6に示したステップ#200における処理の一部であって、図9に示したのとは別の部分の動作を示すフローチャートである。

【符号の説明】

G P ゲーム装置

100 プログラム源

11 コネクタ

12 フラッシュメモリ

13 SRAM

14 メモリコントローラ

15 バックアップIC

DB1 データバス

AB1、AB2 アドレスバス

CE1、CE2 チップセレクト信号

DS デコード信号

200 ゲーム機

62 PPU

63 ワーキングRAM

64 ビデオRAM

65 I/Oインターフェイス

66 コネクタ

70 コントローラ

80 表示器

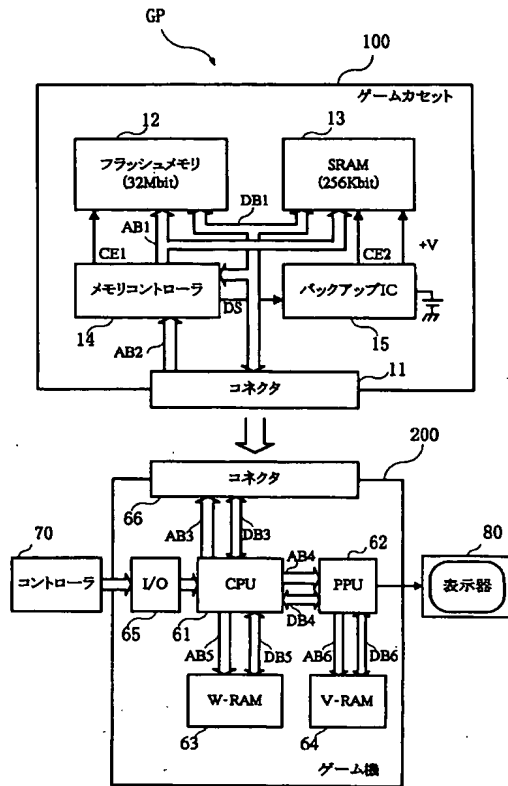
【図3】

ゲーム	シナリオユニット	選択条件
A	a1, a2	無条件
	a3	初めてのプレイ
	a4	ゲームBのみクリア
	a5	ゲームCのみクリア
	a6	ゲームB、Cの順にクリア
	a7	ゲームC、Bの順にクリア
	a8	無条件
B	b1, b2	無条件
	b3	初めてのプレイ
	b4	ゲームAのみクリア
	b5	ゲームCのみクリア
	b6	ゲームA、Cの順にクリア
	b7	ゲームC、Aの順にクリア
	b8	無条件
C	c1, c2	無条件
	c3	初めてのプレイ
	c4	ゲームAのみクリア
	c5	ゲームBのみクリア
	c6	ゲームA、Bの順にクリア
	c7	ゲームB、Aの順にクリア
	c8	無条件

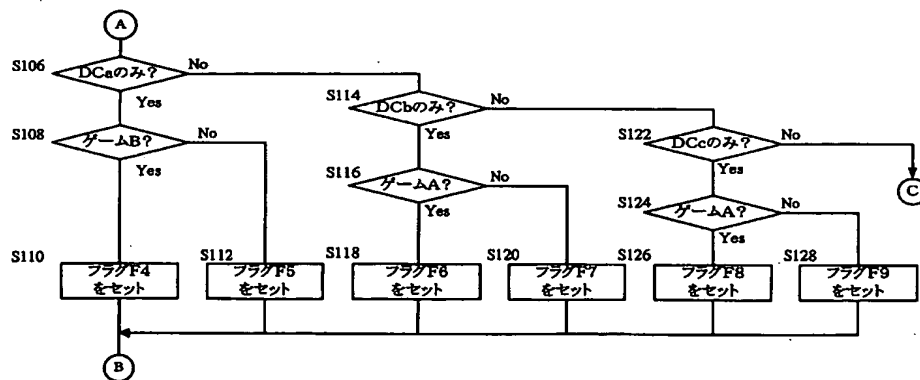
【図4】

ゲーム	発生条件	シナリオ	シナリオユニット	フラグ
A	初めてのプレイ	SAα	a1, a2, a3, a8	F1
	ゲームBのみクリア	SAβ	a1, a2, a4, a8	F6
	ゲームCのみクリア	SAγ	a1, a2, a5, a8	F8
	ゲームB、Cの順にクリア	SAδ	a1, a2, a6, a8	F14
	ゲームC、Bの順にクリア	SAε	a1, a2, a7, a8	F15
B	初めてのプレイ	SBα	b1, b2, b3, b8	F2
	ゲームAのみクリア	SBβ	b1, b2, b4, b8	F4
	ゲームCのみクリア	SBγ	b1, b2, b5, b8	F9
	ゲームA、Cの順にクリア	SBδ	b1, b2, b6, b8	F12
	ゲームC、Aの順にクリア	SBε	b1, b2, b7, b8	F13
C	初めてのプレイ	SCα	c1, c2, c3, c8	F3
	ゲームAのみクリア	SCβ	c1, c2, c4, c8	F5
	ゲームBのみクリア	SCγ	c1, c2, c5, c8	F7
	ゲームA、Bの順にクリア	SCδ	c1, c2, c6, c8	F10
	ゲームB、Aの順にクリア	SCε	c1, c2, c7, c8	F11

【図1】

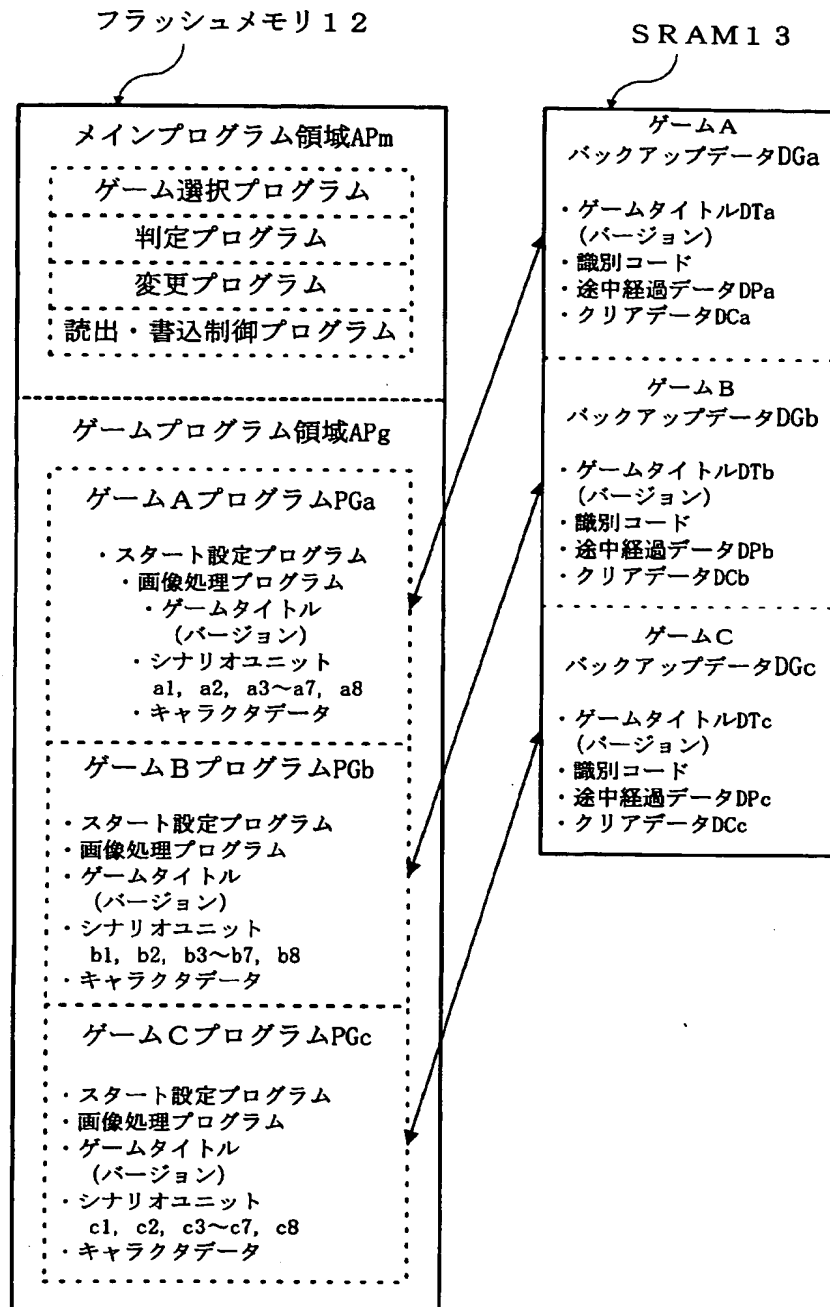


【図7】

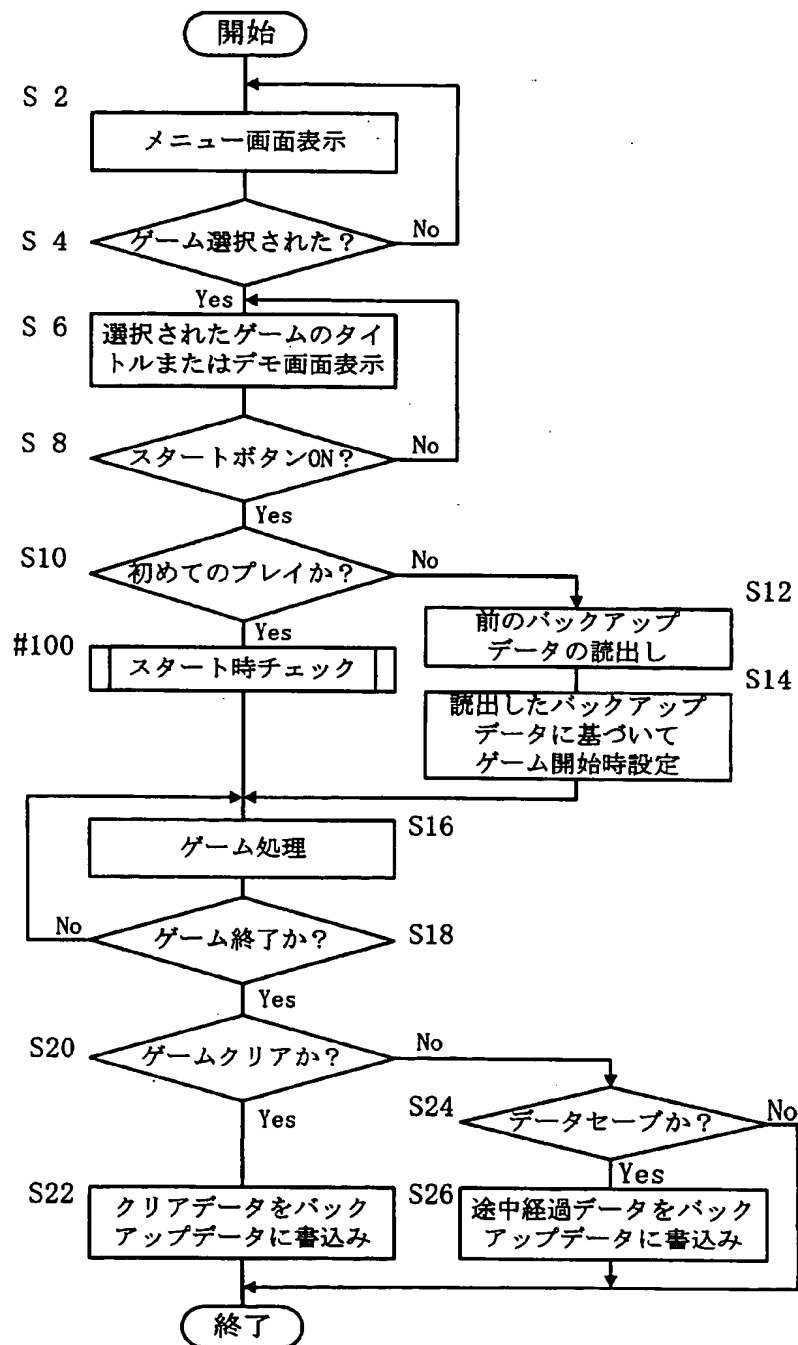




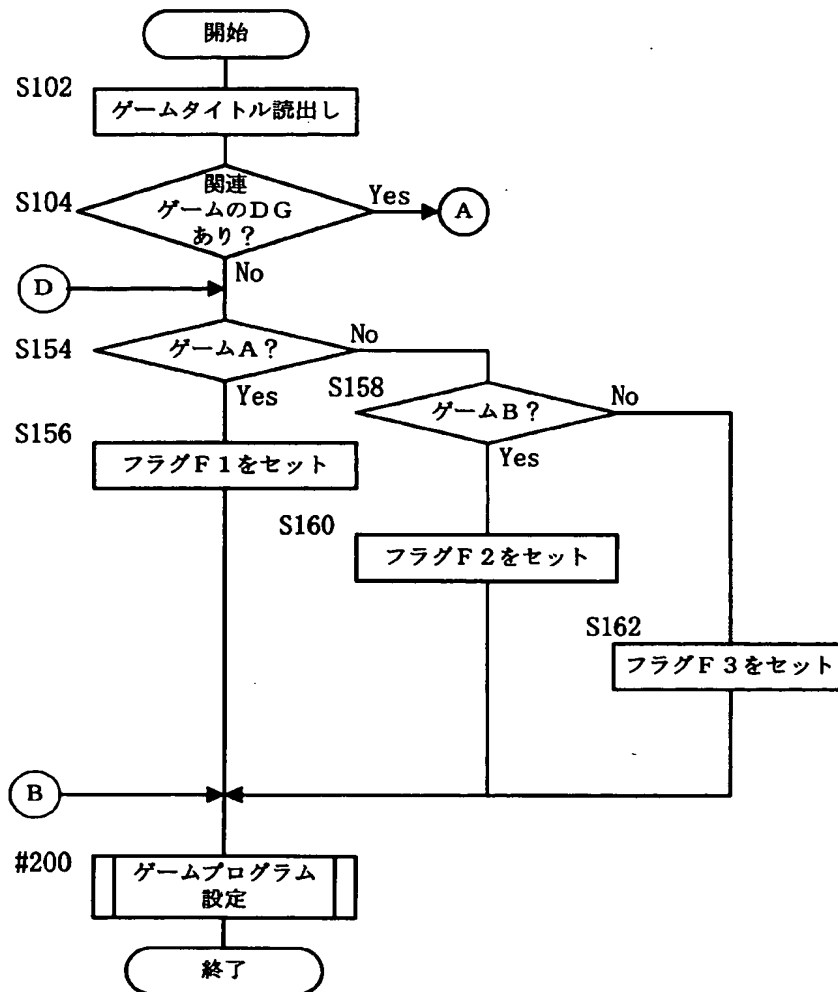
【図2】



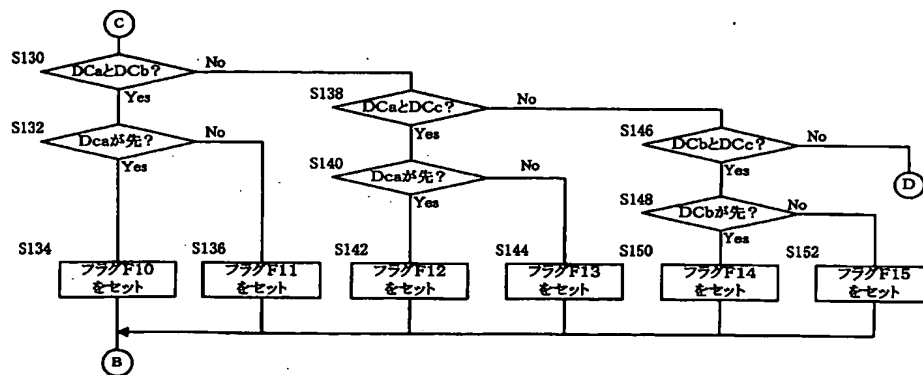
【図5】



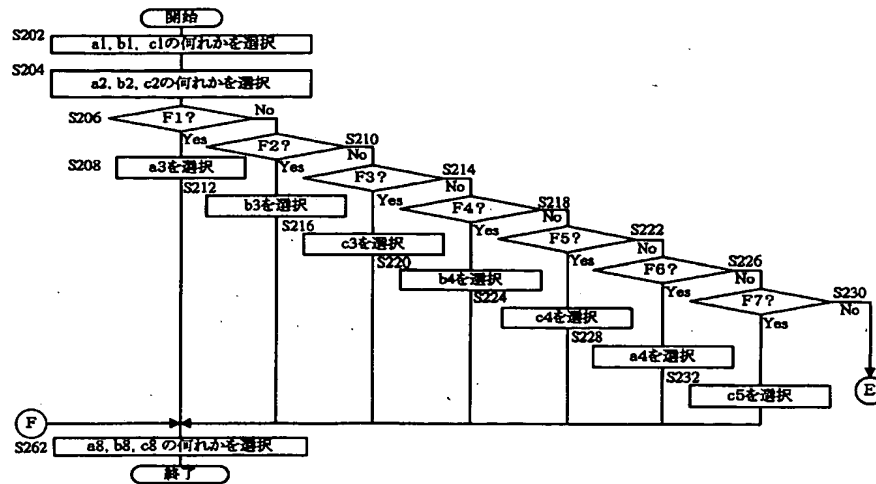
【図6】



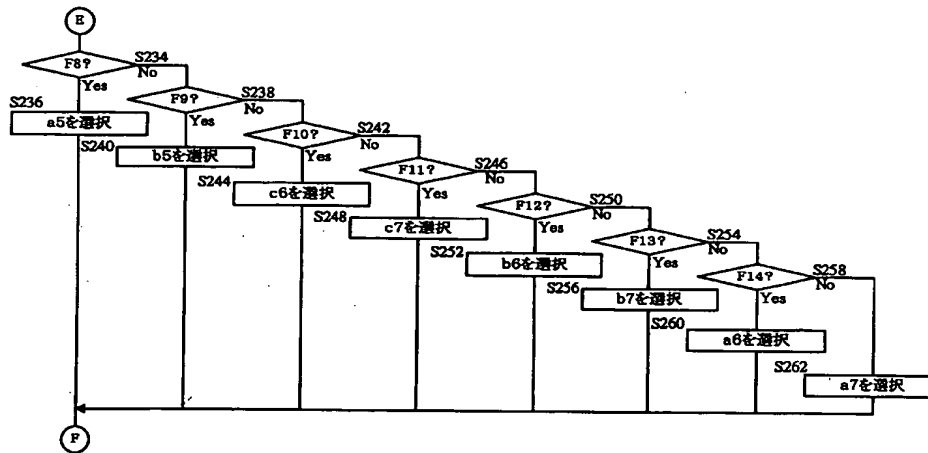
【図8】



【図9】



【図10】



フロントページの続き

(72)発明者 大原 貴夫  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

(72)発明者 吉野 元文  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

(72)発明者 吉岡 照幸  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

(72)発明者 新井 重伸  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

(72)発明者 堀田 拓司  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

(72)発明者 平井 和一  
京都府京都市東山区福稲上高松町60番地  
任天堂株式会社内

Fターム(参考) 2C001 BB00 BB07 BB08 BC00 BC10  
CB01 CB02 CB03 CC02

BEST AVAILABLE COPY